

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) In a computing system which includes a visual design surface in the form of a user interface having a plurality of shapes that are selectable by a user, each shape being associated with one or more configuration parameters that define characteristics of software components such as relative positioning of shapes, connections between shapes and other design parameters set by a user, and wherein a user may select and arrange a plurality of the shapes when designing a software application by dragging and dropping them to a design section of the interface. A method of alerting a user to configuration errors that may arise due to inconsistencies between the configuration parameters of one or more of the selected shapes for the software design of shapes representing software artifacts and displayed on a visual design surface, the method comprising:
 - the user selecting a first shape and copying it to the design section of the interface;
 - the user selecting a second shape and copying it to the design section of the interface so that the second shape is functionally interactive with the first shape;
 - automatically evaluating with a configuration module the configuration parameters of the first and second shapes by accessing a configuration rules database which is used by the configuration evaluation module to determine whether the configuration parameters of the two functionally interactive shapes in the proposed design violate any configuration rules; and
 - when a configuration rule is determined to have been violated, automatically evaluating any errors using an error module that accesses a common error database which defines at least some errors as being grouped under a common error, and then returning a common error for display to the user in order to limit the number of error messages presented to the user during the design process;
- (a) displaying an icon next to a at least one of the first or second selected shapes to represent at least one common configuration error associated with the shape; and
- (b) in response to a user selecting the icon, displaying at least one proposed solution to a the common configuration error presented.

2. (Canceled)
3. (Currently Amended) The method of claims 21 or 31, wherein the configuration parameter rules are selected based on a context in which the shapes ~~is~~are being used.
4. (Currently Amended) The method of claims 21 or 31, ~~wherein (e) is performed after (a) and~~ further including:
 - (d) removing the icon when the user has taken one or more actions and the common configuration error no longer exists.
5. (Canceled)
6. (Canceled)
7. (Canceled)
8. (Currently Amended) The method of claims 1 or 31, wherein the at least one proposed solution ~~comprises~~is presented in a dialog box associated with the icon.
9. (Original) The method of claim 8, wherein the dialog box prompts a user to initialize a variable.
10. (Original) The method of claim 8, wherein the dialog box prompts a user to set a configuration parameter.
11. (Currently Amended) The method of claims 1 or 31, wherein the at least one proposed solution comprises a wizard.
12. (Currently Amended) The method of claims 1 or 31, wherein the at least one proposed solution comprises creating a new design element.

13. (Currently Amended) The method of claims 1 or 31, wherein the at least one proposed solution comprises adding a shape.
14. (Currently Amended) The method of claims 1 or 31, wherein the at least one proposed solution to the common configuration error comprises adding a necessary shape that is not yet included in the design section ~~connected to the shape in (a).~~
15. (Currently Amended) The method of claims 1 or 31, wherein the at least one common configuration error comprises configuration parameters that are set by the user in an inconsistent manner.
16. (Original) The method of claim 15, wherein the inconsistent configuration parameters are configuration parameters of the same shape.
17. (Currently Amended) The method of claim 15, wherein the inconsistent configuration parameters are configuration parameters of at least two the first and second shapes.
18. (Currently Amended) A method of ~~alerting a user of configuration errors of shapes representing software artifacts and displayed on a visual design surface, the method comprising as defined in claims 1 or 31 wherein at least one of the shapes is a container shape~~
(a) ~~displaying an icon next to a container shape to represent at least one configuration error with respect to a shape contained within the container shape; and~~
(b) ~~in response to a user selecting the icon, displaying at least one proposed solution to the configuration error.~~
19. (Currently Amended) The method of claim 18, further including:
(e) (a) expanding the container shape to display at least the shape contained within the container shape; and
(f) (b) displaying the icon next to a shape contained within the container shape and that contains the at least one common configuration error.

20. (Currently Amended) The method of claim 18, wherein the at least one common configuration error comprises a necessary shape that is not connected to the shape contained within the container shape.

21. (Currently Amended) The method of claim 18, wherein the at least one common configuration error comprises configuration parameters set by the user in an inconsistent manner.

22. (Original) The method of claim 21, wherein the inconsistent configuration parameters are configuration parameters of the same shape.

23. (Currently Amended) The method of claim 21, wherein the inconsistent configuration parameters are configuration parameters of the first and second ~~at least two~~ shapes.

24. (Canceled)

25. (Canceled)

26. (Canceled)

27. (Canceled)

28. (Canceled)

29. (Canceled)

30. (Canceled)

31. (New) In a computing system which includes a visual design surface in the form of a user interface having a plurality of shapes that are selectable by a user, each shape being associated with one or more configuration parameters that define characteristics of software components such as relative positioning of shapes, connections between shapes and other design parameters set by a user, and wherein a user may select and arrange a plurality of the shapes when designing a software application by dragging and dropping them to a design section of the interface, a computer program product for implementing a method of alerting the user to configuration errors that may arise due to inconsistencies between the configuration parameters of one or more of the selected shapes for the software design, the computer program product comprising:

a computer readable medium storing executable instructions for implementing the method, and

wherein the method is comprised of:

the user selecting a first shape and copying it to the design section of the interface;

the user selecting a second shape and copying it to the design section of the interface so that the second shape is functionally interactive with the first shape;

automatically evaluating with a configuration module the configuration parameters of the first and second shapes by accessing a configuration rules database which is used by the configuration evaluation module to determine whether the configuration parameters of the two functionally interactive shapes in the proposed design violate any configuration rules; and

when a configuration rule is determined to have been violated, automatically evaluating any errors using an error module that accesses a common error database which defines at least some errors as being grouped under a common error, and then returning a common error for display to the user in order to limit the number of error messages presented to the user during the design process;

displaying an icon next to a at least one of the first or second selected shapes to represent at least one common configuration error associated with the shape; and
in response to a user selecting the icon, displaying at least one proposed solution to the common configuration error presented.